최적 통신 걸침 나무 문제해결을 위한 유전알고리즘의 유전자 표현법에 대한 연구

김종율
부산광역시 사상구 주례2동 산69-1 동서대학교 컴퓨터정보공학부
xmaskjr@gdsu.dongseo.ac.kr

Study on Gene Representation in GA for Optimal Communication Spanning Tree Problems

Jong Ryul Kim
Division of Computer & Information Engineering, Dongseo University

요 약

통신 시스템에 대한 관심은 인터넷의 급격한 발전에 의해 가상공간의 출현과 유비쿼터스 컴퓨팅 환경 구축에 대한 요구가 증대됨에 따라 관련 이론 및 기술의 발전을 주도해 왔다. 이와 관련한 문제들 중에 가장 근간이 되는 문제들 중 하나는 최적 통신 걸침 나무 (OCST: Optimal Communication Spanning Tree) 문제이다. 본 논문에서는 이러한 최적 통신 걸침 나무 문제를 해결하기 위해 유전 알고리즘 (GA)를 이용한다. 유전 알고리즘을 이용함에 있어서 중요한 단계중 하나는 유전자표현을 어떻게 문제에 적합하게 설계하느냐이다. 본 논문에서는 걸침나무를 표현하기 위해 기존의 Prüfer수 기반의 유전자 표현법을 개선하여 n개의 노드에 대해 $n-2$개의 숫자열로 표현가능하면서도 보다 더 최적 통신 걸침 문제에 적합하도록 고안한 새로운 유전자 표현법을 이용한다. 임의로 생성된 예제에 대한 수치 실험을 통해 통신시스템의 기본 문제 중 하나인 최적 통신 걸침 문제의 해법으로서의 제안 알고리즘의 유용성과 효율성을 확인한다.

1. 서 론

인터넷의 출현은 인류에게 많은 도전과 많은 삶의 변화를 주도해 나가고 있다. 인터넷 시장의 규모가 점차 확대됨에 따라 네트워크를 어떻게 설계할 것인가 또는 어떻게 네트워크 서비스를 운용할 것인가에 대한 다양한 연구가 네트워크 설계자, 네트워크 분석가, 네트워크 관리자 등과 같은 많은 관련 연구자들에 의해 폭 넓게 이뤄지고 있다.

일반적으로 컴퓨터 네트워킹, 운송망 최적화 시스템 등과 같은 많은 실세계의 네트워크 설계 문제에서, 어떤 제약조건들을 만족하면서 목적함수들을 최적화하는 네트워크 시스템을 설계하는 것은 매우 중대한 문제 중 하나이다. 이러한 네트워크 설계 문제들 중에 가장 근간이 되는 문제는 최적 걸침 나무 (OCST : Optimum Communication Spanning Tree) 문제이다. 그림 1에서와 같이 OCST를 적용한 대형 통신 네트워크 시스템들은 백본으로 여러 네트워크들을 함께 묶어서 구성된다. 또한 근래에 네트워크를 구축하려고 할 때, 무중계 전송 가능 거리 (최대 120Km 이상), 광대역폭 (거의 50 Tbps), 적은 신호 감쇠 (거의 0.2 dB/km), 적은 신호 왜곡, 낮은 전력 요구 등 [1]과 같은 잠재력을 가지고 있어 광케이블이 주로 사용되고 있으며 광케이블 접속의 비싼 비용을 고려할 때 초기의 네트워크의 구조는 주로 걸침 나무 (Spanning Tree) 형태로 구성되는 것이 일반적이다.

한편 급격히 증가하는 인터넷 이용자에 대응하기 위해서 라우팅 프로토콜 (Routing Protocol)의 제어와 관련한 규모성 문제 (Scalability Issues)를 효과적으로 다룰 수 있어야 하는데 이러한 규모성 문제를 잘 다루기 위해서 ISP (Internet Service Provider)들은 자신이 관리하는 네트워크 (자치 시스템 또는 Autonomous System (AS))를 논리적으로 분할하여 계층적으로 운용하기 시작하였으며 이에 따라 라우팅 프로토콜의 제어는 분할된 각 지역 영역 (Local Area, LA)별로 독립적으로 수행되도록 운용하고 있다. 이러한 계층적 네트워크 시스템은 주로 걸침나무 형태로 구성되고 있다 [2].

또한, 인터넷 수능 방송과 같이 다수의 사용자가 동일한 서비스를 요청할 때 서비스하는 방식인 그룹 통신을 위한 멀티캐스트 서비스에도 그 그룹을 관리하기 위해 주로 걸침나무 형태로 구성하여 서비스하는 것이 일반적이며 유비쿼터스 컴퓨팅을 위해 최근 그 관심이 한층 더 높아지고 있는 센서 네트워크 분야에서도 배터리 운용 등을 고려할 때 현실적으로 가능한 토폴로지는 걸침 나무 형태를 따른다는 연구들이 많이 수행되고 있다 [3].

최적의 OCST를 찾아내는 것은 매우 복잡한 조합최적화 문제이며 NP-hard 최적화 문제로 분류된다. 전통적인 방법들을 이용하면, 네트워크의 규모가 지수적으로 확대되어지는 경우, OCST 문제를 해결해 내기가 힘들어 진다. 이러한 이유 때문에 유전 알고리즘 (GA)을 기초로 하는 메타 휴리스틱 알고리즘이 주목을 받고 있다. GA는 많은 실세계 문제들을 풀기 위해 자주 이용되고 있으며 네트워크 최적화 문제들의 최적화 기술로서의 잠재력을 매우 주목받고 있는 기법이다. 그리고 다목적 문제를 위한 GA 기법으로도 많이 연구되고 있으며 특히 운송망 최적화 문제, 생산프로세스 계획 문제, 최소 걸침 나무 (MST) 문제 등과 같은 문제들에서 다목적 문제에 대한 기법으로 매우 활발한 연구가 이뤄지고 있다 [4, 5].

본 논문에서는 이러한 GA를 이용하여 OCST 문제에 적용가능한 해법을 제안한다. 제안하는 GA는 n개의 노드에 대해 $n-2$개의 숫자열로 표현가능하다고 알려진 기존의 Prüfer수 (PN)를 이용한 표현법을 개선하여 보다 더 실제적인 최적 통신 걸침 문제에 적합하도록 고안한 새로운 유전자 표현법을 이용

한다. 제안하는 새로운 유전자 표현법은 최상위 노드를 포함한 중간 노드들의 연결 상태는 PN을 이용하여 표현하고 그 외 잎 노드와 중간 노드들과의 연결 상태는 클러스트링 문자열로 표현하여 PN만 사용한 알고리즘 보다 더 효율적인 GA를 제안한다. 마지막으로 임의로 생성된 예제를 수치 실험하여 본 논문에서 제안하는 GA가 OCST문제를 효과적이고 효율적으로 풀이할 수 있는 해법임을 확인한다.

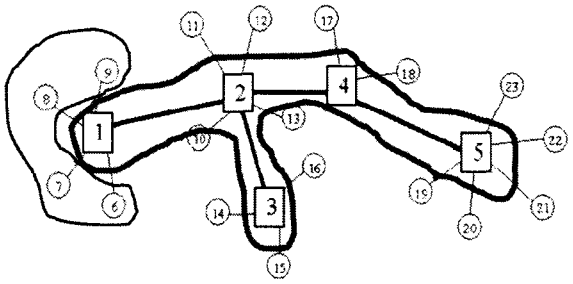


그림 1. 네트워크 구조에 대한 예

2. 최적 걸침 나무 문제

실세계의 네트워크 최적화 설계에 있어서 많은 연구자들에 의해 서로 다른 위치에 있는 n개의 노드를 서로 연결하는 최선의 방법을 제시하는 걸침나무 구조가 가장 통신 네트워크를 설계하기 위해 최선의 토폴로지임이 증명되어 왔다 [6]. 일반적으로 OCST 문제는 전형적인 MST문제와는 달리 각각 노드들의 쌍과 관련이 있는 통신요구량 (Communication requirements)과 각 링크의 연결 비용을 함께 고려한다. 이러한 OCST 문제는 구축비용, 통신 요구량, 네트워크 신뢰도 등과 같은 성능 측도를 최적화하는 최상의 레이아웃 (Layout)을 탐색해 내는 것을 말한다고 할 수 있다. 이러한 시스템의 성능측도들은 매우 중요하며 주로 네트워크 토폴로지에 의해 좌우되고 클러스터링과 라우팅의 두 가지 주요한 항목들도 OCST 문제를 풀어서 얻어진 토폴로지에 의해 결정된다. 클러스터링 문제는 얼마나 많은 중간 내부 노드들 (Intermediate Nodes)로 네트워크가 나눠질 수 있는가와 잎 노드들 (Leaf Nodes)이 클러스터에 연결되어 질 것인가와 같은 두 가지 항목들로 이뤄져 있으며 라우팅 문제는 걸침 나무 토폴로지로 어떻게 상호 연결할 것인가를 결정하는 것이다 [7].

이러한 문제를 설명하기 위해 무향 그래프 $G = (N, L)$를 정의한다. N은 노드 집합을 의미하고 L은 가능한 연결 링크의 집합을 의미하고 T ($\subset G$)는 우리가 찾고자 하는 걸침나무, 즉 트리를 의미한다. 임의의 노드 n과 노드 m이 있을 때, 이 두 노드간의 통신비용 (Communication cost)은 노드 n과 노드 m의 거리와 통신요구량의 곱으로 정의 된다. 노드 n과 노드 m 사이에 경로가 n-i-j-k-m이 존재하고 노드 n과 노드 m 사이의 통신요구량이 q일 경우, 통신비용은 $q \times \{c_{ni}+c_{ij}+c_{jk}+c_{km}\}$ 으로 계산되며 c_{ij}는 노드 i와 노드 j의 연결 비용을 나타낸다. 따라서 본 논문에서 풀이하고자 하는 OCST문제는 수학적으로 다음과 같은 전체 걸침나무의 통신 비용을 최소화 하는 문제로 정의할 수 있다.

$$\min f(T) = \sum_{i,\ j \in N} R(i,\ j) \cdot d_t\ (i,\ j) \qquad (1)$$

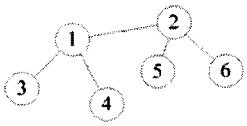
이러한 OCST 문제의 보다 현실적인 문제는 계층적 구조로 되어 있어서 먼저 내부 노드들을 구성하고 잎 노드들이 연결되는 형태로 네트워크 토폴로지들이 이뤄진다.

3. 유전 알고리즘

본 논문에서 제안하는 유전 알고리즘은 실세계의 OCST 문제에 보다 적합한 형태의 유전자 표현법을 이용하여 보다 더 효율적이고 효과적인 OCST 문제에 대한 해법을 제시한다.

3.1 유전자 표현법 및 초기화

유전자 표현은 코딩 공간에서 문제의 후보해를 표현하기 위한 자료구조를 의미한다. 보통 다른 종류의 문제들은 다른 자료구조 또는 유전자 표현을 사용하여야 유전자 알고리즘의 탐색 능력을 훼손하지 않으면서 준 최적해 또는 최적해를 빠르게 탐색해 나갈 수 있다. 걸침 나무로 구성되는 네트워크의 구조를 유전자 표현으로 나타내기 위한 방법은 일반적으로 다음의 두 가지가 있다. 그중 하나는 링크를 기반으로 하는 방법 [8]이고, 또 다른 하나는 *Prüfer* 수 (PN)를 기반으로 하는 방법 [9]이다. 링크를 기반으로 하는 표현 방법은 n개의 노드에 대하여 $n \times (n\text{-}1)/2$의 메모리를 필요로 하지만 PN을 기반하는 표현 방법은 n-2개의 메모리만으로도 걸침 나무를 표현 할 수 있다. 두 방법에 대한 비교 연구는 참고문헌 [10]에서 참조할 수 있으며 결론적으로 말하자면 PN을 기반한 표현 방법이 더 우수한 결과를 얻게 한다고 할 수 있다. 본 논문에서는 이러한 PN을 기반한 유전자 표현법을 실세계의 OCST 문제에 보다 더 적합하면서도 PN의 특성을 살린 새로운 유전자 표현법을 제안한다. 즉, 본 논문에서는 중간 노드들을 PN으로 나타내고 잎 노드들을 클러스터 숫자열로 표현하는 새로운 유전자 표현법을 제안한다.



1	1	2	2

Prüfer number

그림 2. 트리와 그에 대한 Prüfer 수 (PN)

제안하는 유전자 표현법을 설명하기 위해 우선 PN에 대하여 설명하면 다음과 같다. 그래프 이론의 오래된 정리들 중에서 그래프 열거에 대한 일반적인 정리 중 하나는 k개의 노드를 가진 완전 그래프에는 $k^{(k-2)}$개의 걸침나무가 존재한다는 Cayley의 정리이다. Prüfer는 그러한 걸침나무들과 k개의 숫자들로 이뤄지는 k-2길이의 숫자열 집합사이의 일대일 대응관계를 확증함으로써 Cayley의 정리를 증명하였다. 이것은 k개의 다른 숫자들로 구성되는 k-2개의 숫자열로 걸침나무를 표현할 수 있다는 것을 의미한다. 왜냐하면 임의의 걸침나무에 대해 적어도 2개의 잎(leaf) 노드가 존재하기 때문이다. 이러한 숫자 조합의 열을 Prüfer수 (PN)이라고 한다 [4, 5, 10].

예를 들어 설명하자면 그림 2에서 나타낸 것과 같이 6개의 노드로 구성되는 걸침 나무를 PN (1 1 2 2)으로 표현하는 예를 살펴보자. 그림 2에서 먼저 가장 낮은 라벨을 가진 잎노드를 찾는다. 이 경우에는 노드 3에 해당한다. 걸침 나무에서 노드 3은 노드 1에 연결되어 있으므로 PN의 첫숫자로 1을 할당하고 노드 3과 링크 (3, 1)를 트리에서 제거한다. 이런 과정을 링크 (2, 6)이 남을 때 까지 반복하면 그림 2의 걸침 나무에 대한 PN이 생성된다.

반면에 PN으로부터 걸침나무를 생성하는 디코딩 과정을 살

펴보면, P=(1 1 2 2)에 대해 노드 3, 4, 5, 6이 PN에 나타나지 않는 노드 집합을 P'={3, 4, 5, 6}로 나타낼 수 있다. 노드 3이 제일 낮은 라벨을 가진 노드이고 P의 가장 왼쪽에 있는 숫자가 1이므로 노드 3과 노드 1을 연결한 후 P'에서 노드 3을 PN에서 노드 1을 제거한다. 그 다음으로 노드 4와 P의 가장 왼쪽 숫자 1을 고려하여 노드 4와 노드 1을 연결한 후 각각 제거를 한다. 이때 P에서 숫자 1을 제거할 때 이제 더 이상 숫자 1이 나타나지 않기에 숫자1을 집합 P'에 포함시킨다. 그럼, P'는 {1, 5, 6}이 된다. 이러한 과정을 P가 비어지고 P'에 두 개의 숫자가 남을 때까지 반복하고 P'에 남은 두 숫자의 노드를 서로 연결하면 그림 2의 결침나무가 생성된다.

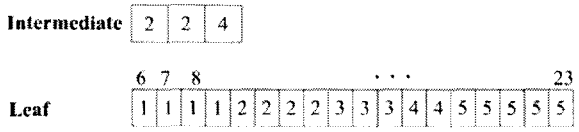


그림 3. 그림 1에 대한 제안한 유전자 표현법의 예

지금까지 PN 인코딩 방법에 대해서 소개하였다. 본 논문에서는 PN의 장점을 살리면서 보다 더 실세계의 OCST 네트워크 토폴로지 설계문제에 적합한 유전자 표현을 위해 다음과 같은 두개의 유전자형을 가지는 인코딩 방법을 제안한다. 중간노드들의 연결 상태는 PN으로 표현하고 중간 노드와 잎 노들사이의 연결 상태들은 클러스트링 문자열으로 나타내는 것이다. 따라서 그림 3에 표시된 것과 같이 초기화 과정에서 서비스 센터는 영역 [1, n]에서 임의로 n-2개의 숫자 (PN)로 생성하여 나타내고 사용자는 영역 [1, n]에서 임의로 m개 숫자 (클러스터 문자열)로 생성한다. 여기서 클러스터 문자열은 어떻게 사용자를 서비스 센터에 할당되는지에 대한 정보를 나타낸 것으로 사용자가 어떤 특정 서비스 센터에 속하는 것을 의미한다. 본 논문에서의 유전자 표현은 n+m-2 길이를 가진다.

3.2 평가 함수

높은 신뢰도를 가지는 네트워크의 토폴로지를 설계하는 문제를 위한 평가함수 $eval(v_k)$의 값을 평가하기 위해 우선 인코딩화된 염색체 v_k를 인접행렬 x_k로 디코딩한다. 그런 후 목적함수 값을 구해 내고 평가함수 $eval(v_k)$을 다음과 같이 계산한다.
여기서 BIG는 매우 큰 수를 나타내며 δ는 만일 $R(x_k)<R_{min}$이면 0이고 그렇지 않으면 1로 설정된다. 그리고 각 세대마다 최고 큰 적합도를 가지는 최선의 염색체 v^*는 다음의 수식을 통해 유지해 나간다.

$$v^* = \text{argmax}\{eval(v_k) \mid k = 1,2, \ldots, pop_size\} \quad (2)$$

여기에서 argmax는 최대값을 가지는 인수(argument of the maximum)를 의미한다.

3.3 선택

GA에서 선택연산은 매우 중요한 역할을 한다. 교차 및 돌연변이 같은 유전자 연산들은 해 공간에서의 탐색을 위한 탐험(exploration)연산이라고 한다면 선택연산은 진화과정을 인도하는 GA를 위한 개척 (exploitation)연산이라고 할 수 있다. 본 논문에서 사용된 선택 연산은 자유로이 해 공간을 탐색하도록 하기 위해 토너먼트 선택법과 엘리티스트 (elitist)방법을 혼합한 형태이다.

토너먼트 선택은 n개의 염색체를 임의로 선택하고 이들 중에서 가장 적응도가 높은 개체를 다음세대의 집단에 포함시킨다. 다음 세대의 염색체들을 모두 얻을 때까지 이 과정을 반복한다. 이때 선택된 염색체를 현재 집단에서 제거하지 않는다. 즉, 한 염색체가 여러 번 선택될 수 있다. 본 논문에서는 n을 5로 설정하여 이용한다. 선택 압력은 토너먼트 크기인 n의 값에 따라 좌우되는데 n이 크면 적합도가 낮은 염색체들이 다음 세대에서 선택되는 기회가 적어진다. 이러한 토너먼트 선택은 최근에 주로 많이 사용되는 선택법 중에 하나이고 구현이 쉬우며 병렬 구조로 수행될 때 용이하고 선택 압력을 쉽게 조절할 수 있다는 장점을 가지고 있다. 본 논문에서 사용하는 선택법을 다음의 프로시저로 설명할 수 있다.

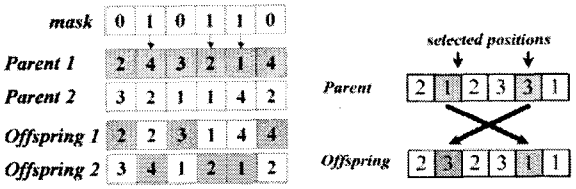
프로시저: 선택법

Step 1: 먼저 엘리티스트 (elitist)방법으로 이전 세대의 집단의 최선해를 다음세대의 집단으로 선택한다.
Step 2: 먼저 이전 세대의 집단 중에서 n개를 선택한다.
Step 3: 선택된 n개 중에서 가장 높은 적합도록 가진 하나의 염색체를 선택한다.
Step 4: 다음 세대의 염색체들을 모두 얻을 때까지 Step 2와 3을 반복한다.

이런 선택연산을 이용하여 최선의 염색체를 현재의 세대에서 다음세대로 유지할 수 있다.

3.4 유전자 연산자들

그림 4에서와 같이 균등교차 (uniform crossover)이라고도 불리는 다점 교차(multi-point crossover)를 이용한다. 이런 형태의 교차 연산은 임의로 두 부모 염색체를 선택한 후 두 부모 염색체 중에 임의의 위치에 있는 유전자를 자식 염색체의 유전자로 복사하여 자식 염색체를 생성해 나간다. 이러한 균등교차는 우선 교차를 위해 임의로 0 또는 1의 마스크 열을 염색체의 길이만큼 생성한 후 한쪽 부모 염색체로(Parent1)부터 마스크가 0인 유전인자를, 다른 쪽 부모 염색체(Parent2)로부터는 마스크가 1인 유전인자를 복사해 하나의 자식 염색체(Offspring1)를 생성하고 다른 자식 염색체(Offspring2)는 앞서와 반대로 하여 생성하는 유전 연산자이다.



(a) 다점 교차 **(b) 교환 돌연변이**

그림 4. 유전 연산자들

본 논문에서의 돌연변이 연산은 그림 4에서 나타낸 것과 같이 하나의 염색체에서 임의로 선택된 두개의 위치에 있는 유전자 들을 교환하는 교환 돌연변이를 사용한다. 즉, 교환 돌연변이는 먼저 염색체 길이의 범위 내에서 임의의 두 위치를 선택한 후, 선택되지 않은 나머지 위치의 유전자들은 부모 염색체로부터 그대로 복사하고 선택된 두 위치의 서로 위치를 바꾸어 복사하여 자식 염색체를 생성하도록 하는 연산자이다.

4. 수치 실험

수치 실험을 통해 본 논문에서 제안하는 새로운 방식의 유전자 표현법을 이용한 유전 알고리즘의 성능에 대한 우수성을 나

타낸다. 수치실험은 다음과 같은 5가지 수치예제로 구성하여 수행하였다:

- 5 (최상위 및 중간노드 수: 3, 잎노드: 2)개의 노드를 가진 문제(Example 1)
- 12 (최상위 및 중간노드 수: 4, 잎노드: 8)개의 노드를 가진 문제(Example 2)
- 15 (최상위 및 중간노드 수: 5, 잎노드: 10)개의 노드를 가진 문제(Example 3)
- 20 (최상위 및 중간노드 수: 6, 잎노드: 14)개의 노드를 가진 문제(Example 4)
- 30 (최상위 및 중간노드 수: 6, 잎노드: 24)개의 노드를 가진 문제(Example 5).

각각은 집단의 크기를 100, 최대 세대수를 1000, 교배율을 0.7, 돌연변이율을 0.6으로 설정하여 수행되었다.

Example 1의 통신 요구량 행렬과 비용행렬은 참고문헌 [6]에 나타내져 있는 것을 이용하였고 Example 2~5의 통신 요구량 행렬과 비용행렬은 각각 [0, 200]사이의 영역에서 임의로 생성한 행렬을 이용하였다. 수치 실험은 각 예제에 대해 20번씩 수행하였으며 최선, 최악의 결과와 20회 실험에 대한 평균을 내어 그 결과들을 표 1에서 요약하여 나타내었다.

표 1에서 볼 수 있듯이 제안한 새로운 유전자 표현법이 PN만 사용한 표현법보다 계산시간은 작거나 거의 비슷하게 요구하면서 더 최적해에 가까운 값들을 찾아 낼 수 있다는 것을 확인할 수 있다.

표 1. 수치실험의 결과

수치예	제안한 유전자 표현법		Prüfer 수를 기반한 유전자 표현법	
	목적함수치 (avg.)	time (sec.)	목적함수치 (avg.)	time (sec.)
1	283	1	285	1.05
2	31348	19.1	32354.4	19.9
3	41,035	40.15	46,539.3	39.7
4	77,949.9	120.9	84,648.3	119.9
5	3,266,463	514.55	3,363,887	560.45

avg.: average, sec.: second, time: computational time

5. 결론

본 논문은 실세계의 네트워크 문제들 중에 가장 기본이 되는 OCST 문제를 위해 새로운 유전자 표현법을 이용하는 유전 알고리즘을 제안하였다. 본 논문에서 제안하는 새로운 유전자 표현법은 n개의 노드를 가진 그래프에 대해서 n-2개의 숫자열을 이용하여 트리를 표현할 수 있는 PN을 이용한 유전 알고리즘 보다 계산시간과 탐색해낸 해의 최적성의 측면에서 더 좋은 성능을 가진 유전자 표현법으로 최상위 노드를 포함한 중간 노드들의 연결 상태는 PN을 이용하여 표현하고 그 외 잎 노드와 중간 노드들과의 연결 상태는 클러스트링 문자열로 표현하여 PN만 사용한 알고리즘 보다 더 효율적인 GA를 제안한다. 제안한 유전자 표현법을 이용한 수치 실험을 통해 그에 대한 증명을 하였다. 앞으로의 연구는 보다 더 많은 수치예제에 적용해 보고 메시지 지연시간 및 네트워크 신뢰도에 대한 제약이 있는 OCST 문제에 대한 해법을 고안하는 것에 열중할 것이다.

참고문헌

[1] Mukherjee, B., *Optical Communication Networks*, McGraw-Hill, Inc., New York, 1997.

[2] 김도훈, "계층화된 링크-상태 인터넷 라우팅에서 가상 링크 운용 최적화를 위한 다기준 유전자 알고리즘의 응용", *IE Interfaces*, Vol. 16, Special Edition, pp. 14-20, December 2003.

[3] T. Chen, H. Tsai, and C. Chu, "Gathering-Load-Balanced Tree Protocol for Wireless Sensor Networks," *IEEE International Conference on Sensor Networks*, Ubiquitous, and Trustworthy Computing, vol.2, pp.8-13, 2006.

[4] Gen, M. and R. Cheng, *Genetic Algorithms and Engineering Design*, John Wiley & Sons, New York, 1997.

[5] Gen, M. and R. Cheng, *Genetic Algorithms and Engineering Optimization*, John Wiley & Sons, New York, 2000.

[6] S. Soak, S. Chang, S. Byun, and B. Ahn, "Evolutionary Algorithm for Solving Optimum Communication Spanning Tree Problem," Journal of the Korean Information Science Society: SW and Applications, vol.32, no.4, pp.268-276, 2005 (in Korean).

[7] J. R. Kim and M. Gen, "A Genetic Algorithm for Solving Bicriteria Network Topology Design Problems", *Journal of Japan Society for Fuzzy Theory and Systems*, Vol.12, No.1, pp.43-54, 2000.

[8] Elbaum, R. and M. Sidi, "Topological design of local-area networks using genetic algorithms," *IEEE/ACM Transactions on Networking*, Vol.4, No.5, pp.766-778, 1996.

[9] H. Prüfer, "Neuer beweis eines satzesüber permutationen," Arch. Math. Phys, vol.27, pp.742-744, 1918.

[10] 김종율, 이재욱, 현광남, "2기준 네트워크 설계를 위한 새로운 인코딩법을 기반으로 하는 유전자 알고리즘", 정보과학회 논문지: 소프트웨어 및 응용, 제32권, 제10호,pp.963-973, 2005.10.

[11] Bertsekas, D. and R. Gallager, *Data Networks*, 2nd ed., Prentice-Hall, New Jersey, 1992.

[12] Skiena, S., *Implementing Discrete Mathematics Combinatorics and Graph Theory with Mathematica*, Addison-Wesley, Reading, MA, 1990.

[13] Kershenbaum, A. and R. Van Slyke, "Recursive analysis of network reliability," *Networks*, Vol.3, pp.81-94, 1973.

[14] G. R. Raidl and B. A. Julstrom, "Edge sets: an effective evolutionary coding of spanning trees," *IEEE Transaction on Evolutionary Computation*, Vol.7, No.3, pp.225-239, June, 2003.

[15] Gen, M. and R. Cheng, "Evolutionary network design: hybrid genetic algorithms approach," *International Journal of Computational Intelligence and Applications*, Vol.3, No.4, pp.357-380, 2003.

[16] A. Kershenbaum, Telecommunication Network Design Algorithms, McGraw-Hill, Inc., New York, 1993.

[17]G. Syswerda, "Uniform crossover in genetic algorithms," *Proceedings of the Third International Conference on Genetic Algorithms*, pp.2-9, 1989.